

GLOBAL
EDITION

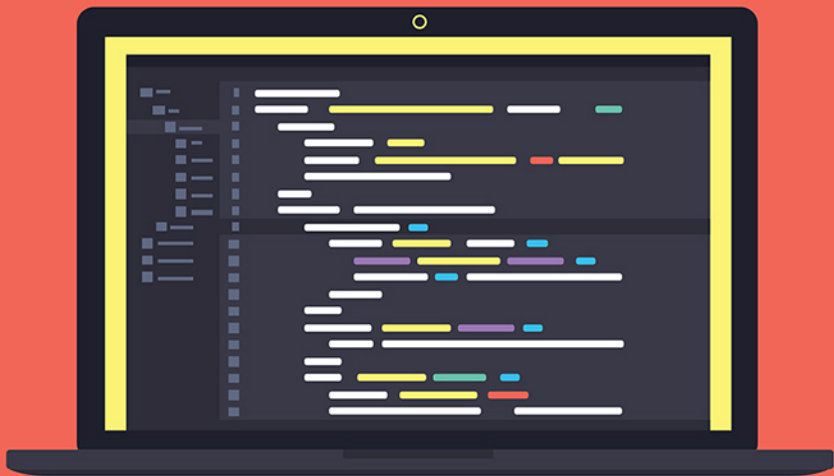


Java™ How to Program

Late Objects

ELEVENTH EDITION

Paul Deitel • Harvey Deitel



DIGITAL RESOURCES FOR STUDENTS

Your new textbook provides 12-month access to digital resources that may include VideoNotes (step-by-step video tutorials on programming concepts), source code, web chapters, quizzes, and more. Refer to the preface in the textbook for a detailed list of resources.

Follow the instructions below to register for the Companion Website for Paul Deitel and Harvey Deitel's **Java™ How to Program, Late Objects, Eleventh Edition, Global Edition**.

- 1 Go to **www.pearsonglobaleditions.com/deitel**.
- 2 Enter the title of your textbook or browse by author name.
- 3 Click Companion Website.
- 4 Click Register and follow the on-screen instructions to create a login name and password.

ISSLDO-WHIFF-SAURY-LAMBS-DOLBY-LIKES

Use the login name and password you created during registration to start using the digital resources that accompany your textbook.

IMPORTANT

This prepaid subscription does not include access to MyProgrammingLab, which is available at **www.myprogramminglab.com** for purchase.

This access code can only be used once. This subscription is valid for 12 months upon activation and is not transferable.

For technical support go to **<https://support.pearson.com/getsupport>**

Java How To Program, Late Objects, Global Edition

Table of Contents

Cover

Title Page

Copyright Page

Contents

Foreword

Preface

Before You Begin

1 Introduction to Computers, the Internet and Java 53

1.1 Introduction 54

1.2 Hardware and Software 56

1.2.1 Moores Law 56

1.2.2 Computer Organization 57

1.3 Data Hierarchy 59

1.4 Machine Languages, Assembly Languages and High-Level
Languages 61

1.5 Basic Introduction to Object Terminology 62

1.5.1 Automobile as an Object 63

1.5.2 Methods and Classes 63

1.5.3 Instantiation 63

1.5.4 Reuse 63

1.5.5 Messages and Method Calls 64

1.5.6 Attributes and Instance Variables 64

Table of Contents

1.5.7 Encapsulation and Information Hiding	64
1.5.8 Inheritance	64
1.5.9 Interfaces	65
1.5.10 Object-Oriented Analysis and Design (OOAD)	65
1.5.11 The UML (Unified Modeling Language)	65
1.6 Operating Systems	66
1.6.1 WindowsA Proprietary Operating System	66
1.6.2 LinuxAn Open-Source Operating System	66
1.6.3 Apples macOS and Apples iOS for iPhone®, iPad® and iPod Touch® Devices	67
1.6.4 Googles Android	67
1.7 Programming Languages	68
1.8 Java	70
1.9 A Typical Java Development Environment	71
1.10 Test-Driving a Java Application	74
1.11 Internet and World Wide Web	78
1.11.1 Internet: A Network of Networks	79
1.11.2 World Wide Web: Making the Internet User-Friendly	79
1.11.3 Web Services and Mashups	79
1.11.4 Internet of Things	80
1.12 Software Technologies	81
1.13 Getting Your Questions Answered	83
2 Introduction to Java Applications; Input/Output and Operators	87
2.1 Introduction	88
2.2 Your First Program in Java: Printing a Line of Text	88
2.2.1 Compiling the Application	92
2.2.2 Executing the Application	93

Table of Contents

2.3	Modifying Your First Java Program	94
2.4	Displaying Text with printf	96
2.5	Another Application: Adding Integers	97
2.5.1	import Declarations	98
2.5.2	Declaring and Creating a Scanner to Obtain User Input from the Keyboard	98
2.5.3	Prompting the User for Input	99
2.5.4	Declaring a Variable to Store an Integer and Obtaining an Integer from the Keyboard	99
2.5.5	Obtaining a Second Integer	100
2.5.6	Using Variables in a Calculation	100
2.5.7	Displaying the Calculation Result	100
2.5.8	Java API Documentation	101
2.5.9	Declaring and Initializing Variables in Separate Statements	101
2.6	Memory Concepts	101
2.7	Arithmetic	102
2.8	Decision Making: Equality and Relational Operators	106
2.9	Wrap-Up	109
3	Control Statements: Part 1; Assignment, ++ and -- Operators	120
3.1	Introduction	121
3.2	Algorithms	121
3.3	Pseudocode	122
3.4	Control Structures	122
3.4.1	Sequence Structure in Java	123
3.4.2	Selection Statements in Java	124
3.4.3	Iteration Statements in Java	124
3.4.4	Summary of Control Statements in Java	124

Table of Contents

3.5 if Single-Selection Statement	125
3.6 ifelse Double-Selection Statement	126
3.6.1 Nested ifelse Statements	127
3.6.2 Dangling-else Problem	128
3.6.3 Blocks	128
3.6.4 Conditional Operator (?:)	129
3.7 while Iteration Statement	129
3.8 Formulating Algorithms: Counter-Controlled Iteration	131
3.9 Formulating Algorithms: Sentinel-Controlled Iteration	135
3.10 Formulating Algorithms: Nested Control Statements	142
3.11 Compound Assignment Operators	146
3.12 Increment and Decrement Operators	147
3.13 Primitive Types	150
3.14 Wrap-Up	150
4 Control Statements: Part 2; Logical Operators	164
4.1 Introduction	165
4.2 Essentials of Counter-Controlled Iteration	165
4.3 for Iteration Statement	166
4.4 Examples Using the for Statement	170
4.4.1 Application: Summing the Even Integers from 2 to 20	171
4.4.2 Application: Compound-Interest Calculations	172
4.5 dowhile Iteration Statement	175
4.6 switch Multiple-Selection Statement	176
4.7 break and continue Statements	182
4.7.1 break Statement	182
4.7.2 continue Statement	182
4.8 Logical Operators	183

Table of Contents

- 4.8.1 Conditional AND (&&) Operator 184
- 4.8.2 Conditional OR (||) Operator 184
- 4.8.3 Short-Circuit Evaluation of Complex Conditions 185
- 4.8.4 Boolean Logical AND (&) and Boolean Logical Inclusive OR (|)
Operators 185
- 4.8.5 Boolean Logical Exclusive OR (^) 186
- 4.8.6 Logical Negation (!) Operator 186
- 4.8.7 Logical Operators Example 187

4.9 Structured-Programming Summary 189

4.10 Wrap-Up 194

5 Methods 204

- 5.1 Introduction 205
- 5.2 Program Units in Java 205
- 5.3 static Methods, static Variables and Class Math 207
- 5.4 Declaring Methods 209
- 5.5 Notes on Declaring and Using Methods 213
- 5.6 Method-Call Stack and Activation Records 214
 - 5.6.1 Method-Call Stack 214
 - 5.6.2 Stack Frames 214
 - 5.6.3 Local Variables and Stack Frames 215
 - 5.6.4 Stack Overflow 215
- 5.7 Argument Promotion and Casting 215
- 5.8 Java API Packages 216
- 5.9 Case Study: Secure Random-Number Generation 218
- 5.10 Case Study: A Game of Chance; Introducing enums 223
- 5.11 Scope of Declarations 227
- 5.12 Method Overloading 230
 - 5.12.1 Declaring Overloaded Methods 230

Table of Contents

5.12.2 Distinguishing Between Overloaded Methods 231

5.12.3 Return Types of Overloaded Methods 231

5.13 Wrap-Up 232

6 Arrays and ArrayLists 245

6.1 Introduction 246

6.2 Primitive Types vs. Reference Types 247

6.3 Arrays 247

6.4 Declaring and Creating Arrays 249

6.5 Examples Using Arrays 250

6.5.1 Creating and Initializing an Array 250

6.5.2 Using an Array Initializer 251

6.5.3 Calculating the Values to Store in an Array 252

6.5.4 Summing the Elements of an Array 253

6.5.5 Using Bar Charts to Display Array Data Graphically 254

6.5.6 Using the Elements of an Array as Counters 256

6.5.7 Using Arrays to Analyze Survey Results 257

6.6 Exception Handling: Processing the Incorrect Response 259

6.6.1 The try Statement 259

6.6.2 Executing the catch Block 259

6.6.3 toString Method of the Exception Parameter 260

6.7 Enhanced for Statement 260

6.8 Passing Arrays to Methods 261

6.9 Pass-By-Value vs. Pass-By-Reference 264

6.10 Multidimensional Arrays 264

6.10.1 Arrays of One-Dimensional Arrays 265

6.10.2 Two-Dimensional Arrays with Rows of Different Lengths 265

6.10.3 Creating Two-Dimensional Arrays with Array-Creation Expressions

266

Table of Contents

6.10.4 Two-Dimensional Array Example: Displaying Element Values	266
6.10.5 Common Multidimensional-Array Manipulations Performed with for Statements	267
6.11 Variable-Length Argument Lists	268
6.12 Using Command-Line Arguments	269
6.13 Class Arrays	271
6.14 Introduction to Collections and Class ArrayList	274
6.15 Wrap-Up	278
7 Introduction to Classes and Objects	298
7.1 Introduction	299
7.2 Instance Variables, set Methods and get Methods	300
7.2.1 Account Class with an Instance Variable, and set and get Methods	300
7.2.2 AccountTest Class That Creates and Uses an Object of Class Account	302
7.2.3 Compiling and Executing an App with Multiple Classes	305
7.2.4 Account UML Class Diagram	305
7.2.5 Additional Notes on Class AccountTest	306
7.2.6 Software Engineering with private Instance Variables and public set and get Methods	307
7.3 Default and Explicit Initialization for Instance Variables	308
7.4 Account Class: Initializing Objects with Constructors	309
7.4.1 Declaring an Account Constructor for Custom Object Initialization	309
7.4.2 Class AccountTest: Initializing Account Objects When Theyre Created	310
7.5 Account Class with a Balance	312
7.5.1 Account Class with a balance Instance Variable of Type double	312
7.5.2 AccountTest Class to Use Class Account	313
7.6 Case Study: Card Shuffling and Dealing Simulation	316

Table of Contents

7.7 Case Study: Class GradeBook Using an Array toStore Grades
320

7.8 Case Study: Class GradeBook Using a Two-Dimensional Array
326

7.9 Wrap-Up 331

8 Classes and Objects: A Deeper Look 339

8.1 Introduction 340

8.2 Time Class Case Study 340

8.3 Controlling Access to Members 345

8.4 Referring to the Current Objects Members with the this Reference
346

8.5 Time Class Case Study: Overloaded Constructors 348

8.6 Default and No-Argument Constructors 353

8.7 Notes on Set and Get Methods 354

8.8 Composition 355

8.9 enum Types 358

8.10 Garbage Collection 361

8.11 static Class Members 361

8.12 static Import 365

8.13 final Instance Variables 366

8.14 Package Access 367

8.15 Using BigDecimal for Precise Monetary Calculations 368

8.16 (Optional) GUI and Graphics Case Study: Using Objects with
Graphics 371

8.17 Wrap-Up 375

9 Object-Oriented Programming: Inheritance 383

9.1 Introduction 384

Table of Contents

9.2 Superclasses and Subclasses 385

9.3 protected Members 387

9.4 Relationship Between Superclasses and Subclasses 388

9.4.1 Creating and Using a CommissionEmployee Class 388

9.4.2 Creating and Using a BasePlusCommissionEmployee Class 393

9.4.3 Creating a CommissionEmployeeBasePlusCommissionEmployee
Inheritance Hierarchy 398

9.4.4 CommissionEmployeeBasePlusCommissionEmployee Inheritance
Hierarchy Using protected Instance Variables 401

9.4.5 CommissionEmployeeBasePlusCommissionEmployee Inheritance
Hierarchy Using private Instance Variables 404

9.5 Constructors in Subclasses 408

9.6 Class Object 409

9.7 Designing with Composition vs. Inheritance 410

9.8 Wrap-Up 412

10 Object-Oriented Programming: Polymorphism and
Interfaces 417

10.1 Introduction 418

10.2 Polymorphism Examples 420

10.3 Demonstrating Polymorphic Behavior 421

10.4 Abstract Classes and Methods 423

10.5 Case Study: Payroll System Using Polymorphism 426

10.5.1 Abstract Superclass Employee 427

10.5.2 Concrete Subclass SalariedEmployee 429

10.5.3 Concrete Subclass HourlyEmployee 431

10.5.4 Concrete Subclass CommissionEmployee 432

10.5.5 Indirect Concrete Subclass BasePlusCommissionEmployee 434

10.5.6 Polymorphic Processing, Operator instanceof and Downcasting 435

Table of Contents

10.6	Allowed Assignments Between Superclass and Subclass Variables	440
10.7	final Methods and Classes	440
10.8	A Deeper Explanation of Issues with Calling Methods from Constructors	441
10.9	Creating and Using Interfaces	442
10.9.1	Developing a Payable Hierarchy	444
10.9.2	Interface Payable	445
10.9.3	Class Invoice	445
10.9.4	Modifying Class Employee to Implement Interface Payable	447
10.9.5	Using Interface Payable to Process Invoices and Employees Polymorphically	449
10.9.6	Some Common Interfaces of the Java API	450
10.10	Java SE 8 Interface Enhancements	451
10.10.1	default Interface Methods	451
10.10.2	static Interface Methods	452
10.10.3	Functional Interfaces	452
10.11	Java SE 9 private Interface Methods	453
10.12	private Constructors	453
10.13	Program to an Interface, Not an Implementation	454
10.13.1	Implementation Inheritance Is Best for Small Numbers of Tightly Coupled Classes	454
10.13.2	Interface Inheritance Is Best for Flexibility	454
10.13.3	Rethinking the Employee Hierarchy	455
10.14	(Optional) GUI and Graphics Case Study: Drawing with Polymorphism	456
10.15	Wrap-Up	458
11	Exception Handling: A Deeper Look	465

Table of Contents

11.1	Introduction	466
11.2	Example: Divide by Zero without Exception Handling	467
11.3	Example: Handling ArithmeticExceptions and InputMismatchExceptions	469
11.4	When to Use Exception Handling	475
11.5	Java Exception Hierarchy	475
11.6	finally Block	479
11.7	Stack Unwinding and Obtaining Information from an Exception	483
11.8	Chained Exceptions	486
11.9	Declaring New Exception Types	488
11.10	Preconditions and Postconditions	489
11.11	Assertions	489
11.12	try-with-Resources: Automatic ResourceDeallocation	491
11.13	Wrap-Up	492
12	JavaFX Graphical User Interfaces: Part 1	498
12.1	Introduction	499
12.2	JavaFX Scene Builder	500
12.3	JavaFX App Window Structure	501
12.4	Welcome AppDisplaying Text and an Image	502
12.4.1	Opening Scene Builder and Creating the File Welcome.fxml	502
12.4.2	Adding an Image to the Folder Containing Welcome.fxml	503
12.4.3	Creating a VBox Layout Container	503
12.4.4	Configuring the VBox Layout Container	504
12.4.5	Adding and Configuring a Label	504
12.4.6	Adding and Configuring an ImageView	505
12.4.7	Previewing the Welcome GUI	507
12.5	Tip Calculator AppIntroduction to Event Handling	507

Table of Contents

12.5.1 Test-Driving the Tip Calculator App	508
12.5.2 Technologies Overview	509
12.5.3 Building the Apps GUI	511
12.5.4 TipCalculator Class	518
12.5.5 TipCalculatorController Class	520
12.6 Features Covered in the Other JavaFX Chapters	525
12.7 Wrap-Up	
13 JavaFX GUI: Part 2	533
13.1 Introduction	534
13.2 Laying Out Nodes in a Scene Graph	534
13.3 Painter App: RadioButtons, Mouse Events and Shapes	536
13.3.1 Technologies Overview	536
13.3.2 Creating the Painter.fxml File	538
13.3.3 Building the GUI	538
13.3.4 Painter Subclass of Application	541
13.3.5 PainterController Class	542
13.4 Color Chooser App: Property Bindings andProperty Listeners	546
13.4.1 Technologies Overview	546
13.4.2 Building the GUI	547
13.4.3 ColorChooser Subclass of Application	549
13.4.4 ColorChooserController Class	550
13.5 Cover Viewer App: Data-Driven GUIs with JavaFX Collections	552
13.5.1 Technologies Overview	553
13.5.2 Adding Images to the Apps Folder	553
13.5.3 Building the GUI	553
13.5.4 CoverViewer Subclass of Application	555
13.5.5 CoverViewerController Class	555

Table of Contents

13.6	Cover Viewer App: Customizing ListView Cells	557
13.6.1	Technologies Overview	558
13.6.2	Copying the CoverViewer App	558
13.6.3	ImageTextCell Custom Cell Factory Class	559
13.6.4	CoverViewerController Class	560
13.7	Additional JavaFX Capabilities	561
13.8	JavaFX 9: Java SE 9 JavaFX Updates	563
13.9	Wrap-Up	565
14	Strings, Characters and Regular Expressions	574
14.1	Introduction	575
14.2	Fundamentals of Characters and Strings	575
14.3	Class String	576
14.3.1	String Constructors	576
14.3.2	String Methods length, charAt and getChars	577
14.3.3	Comparing Strings	579
14.3.4	Locating Characters and Substrings in Strings	583
14.3.5	Extracting Substrings from Strings	585
14.3.6	Concatenating Strings	586
14.3.7	Miscellaneous String Methods	587
14.3.8	String Method valueOf	588
14.4	Class StringBuilder	589
14.4.1	StringBuilder Constructors	590
14.4.2	StringBuilder Methods length, capacity, setLength and ensureCapacity	591
14.4.3	StringBuilder Methods charAt, setCharAt, getChars and reverse	592
14.4.4	StringBuilder append Methods	593
14.4.5	StringBuilder Insertion and Deletion Methods	595
14.5	Class Character	596

Table of Contents

14.6 Tokenizing Strings	601
14.7 Regular Expressions, Class Pattern and Class Matcher	602
14.7.1 Replacing Substrings and Splitting Strings	607
14.7.2 Classes Pattern and Matcher	609
14.8 Wrap-Up	611
15 Files, Input/Output Streams, NIO and XML Serialization	622
15.1 Introduction	623
15.2 Files and Streams	623
15.3 Using NIO Classes and Interfaces to Get File and Directory Information	625
15.4 Sequential Text Files	629
15.4.1 Creating a Sequential Text File	629
15.4.2 Reading Data from a Sequential Text File	632
15.4.3 Case Study: A Credit-Inquiry Program	633
15.4.4 Updating Sequential Files	638
15.5 XML Serialization	638
15.5.1 Creating a Sequential File Using XML Serialization	638
15.5.2 Reading and Deserializing Data from a Sequential File	644
15.6 FileChooser and DirectoryChooser Dialogs	645
15.7 (Optional) Additional java.io Classes	651
15.7.1 Interfaces and Classes for Byte-Based Input and Output	651
15.7.2 Interfaces and Classes for Character-Based Input and Output	653
15.8 Wrap-Up	654
16 Generic Collections	662
16.1 Introduction	663
16.2 Collections Overview	663
16.3 Type-Wrapper Classes	665
16.4 Autoboxing and Auto-Unboxing	665

Table of Contents

16.5 Interface Collection and Class Collections 665

16.6 Lists 666

16.6.1 ArrayList and Iterator 667

16.6.2 LinkedList 669

16.7 Collections Methods 674

16.7.1 Method sort 675

16.7.2 Method shuffle 678

16.7.3 Methods reverse, fill, copy, max and min 680

16.7.4 Method binarySearch 682

16.7.5 Methods addAll, frequency and disjoint 683

16.8 Class PriorityQueue and Interface Queue 685

16.9 Sets 686

16.10 Maps 689

16.11 Synchronized Collections 693

16.12 Unmodifiable Collections 693

16.13 Abstract Implementations 694

16.14 Java SE 9: Convenience Factory Methods for Immutable
Collections 694

16.15 Wrap-Up 698

17 Lambdas and Streams 704

17.1 Introduction 705

17.2 Streams and Reduction 707

17.2.1 Summing the Integers from 1 through 10 with a for Loop 707

17.2.2 External Iteration with for Is Error Prone 708

17.2.3 Summing with a Stream and Reduction 708

17.2.4 Internal Iteration 709

17.3 Mapping and Lambdas 710

17.3.1 Lambda Expressions 711

Table of Contents

17.3.2 Lambda Syntax	712
17.3.3 Intermediate and Terminal Operations	713
17.4 Filtering	714
17.5 How Elements Move Through Stream Pipelines	716
17.6 Method References	717
17.6.1 Creating an IntStream of Random Values	718
17.6.2 Performing a Task on Each Stream Element with forEach and a Method Reference	718
17.6.3 Mapping Integers to String Objects with mapToObj	719
17.6.4 Concatenating Strings with collect	719
17.7 IntStream Operations	720
17.7.1 Creating an IntStream and Displaying Its Values	721
17.7.2 Terminal Operations count, min, max, sum and average	721
17.7.3 Terminal Operation reduce	722
17.7.4 Sorting IntStream Values	724
17.8 Functional Interfaces	725
17.9 Lambdas: A Deeper Look	726
17.10 Stream<Integer> Manipulations	727
17.10.1 Creating a Stream<Integer>	728
17.10.2 Sorting a Stream and Collecting the Results	729
17.10.3 Filtering a Stream and Storing the Results for Later Use	729
17.10.4 Filtering and Sorting a Stream and Collecting the Results	730
17.10.5 Sorting Previously Collected Results	730
17.11 Stream<String> Manipulations	730
17.11.1 Mapping Strings to Uppercase	731
17.11.2 Filtering Strings Then Sorting Them in Case-InsensitiveAscending Order	732
17.11.3 Filtering Strings Then Sorting Them in Case-Insensitive Descending Order	732

Table of Contents

17.12 Stream<Employee> Manipulations 733

17.12.1 Creating and Displaying a List<Employee> 734

17.12.2 Filtering Employees with Salaries in a Specified Range 735

17.12.3 Sorting Employees By Multiple Fields 738

17.12.4 Mapping Employees to Unique-Last-Name Strings 740

17.12.5 Grouping Employees By Department 741

17.12.6 Counting the Number of Employees in Each Department 742

17.12.7 Summing and Averaging Employee Salaries 743

17.13 Creating a Stream<String> from a File 744

17.14 Streams of Random Values 747

17.15 Infinite Streams 749

17.16 Lambda Event Handlers 751

17.17 Additional Notes on Java SE 8 Interfaces 751

17.18 Wrap-Up 752

18 Recursion 766

18.1 Introduction 767

18.2 Recursion Concepts 768

18.3 Example Using Recursion: Factorials 769

18.4 Reimplementing Class FactorialCalculator Using BigInteger 771

18.5 Example Using Recursion: Fibonacci Series 773

18.6 Recursion and the Method-Call Stack 776

18.7 Recursion vs. Iteration 777

18.8 Towers of Hanoi 779

18.9 Fractals 781

18.9.1 Koch Curve Fractal 782

18.9.2 (Optional) Case Study: Lo Feather Fractal 783

18.9.3 (Optional) Fractal App GUI 785

Table of Contents

18.9.4 (Optional) FractalController Class	787
18.10 Recursive Backtracking	792
18.11 Wrap-Up	792
19 Searching, Sorting and Big O	801
19.1 Introduction	802
19.2 Linear Search	803
19.3 Big O Notation	806
19.3.1 $O(1)$ Algorithms	806
19.3.2 $O(n)$ Algorithms	806
19.3.3 $O(n^2)$ Algorithms	806
19.3.4 Big O of the Linear Search	807
19.4 Binary Search	807
19.4.1 Binary Search Implementation	808
19.4.2 Efficiency of the Binary Search	811
19.5 Sorting Algorithms	812
19.6 Selection Sort	812
19.6.1 Selection Sort Implementation	813
19.6.2 Efficiency of the Selection Sort	815
19.7 Insertion Sort	815
19.7.1 Insertion Sort Implementation	816
19.7.2 Efficiency of the Insertion Sort	818
19.8 Merge Sort	819
19.8.1 Merge Sort Implementation	819
19.8.2 Efficiency of the Merge Sort	824
19.9 Big O Summary for This Chapters Searching and Sorting Algorithms	824
19.10 Massive Parallelism and Parallel Algorithms	825
19.11 Wrap-Up	825

Table of Contents

20 Generic Classes and Methods: A Deeper Look 831

20.1 Introduction 832

20.2 Motivation for Generic Methods 832

20.3 Generic Methods: Implementation and Compile-Time
Translation 834

20.4 Additional Compile-Time Translation Issues: Methods That
Use a Type Parameter as the Return Type 837

20.5 Overloading Generic Methods 840

20.6 Generic Classes 841

20.7 Wildcards in Methods That Accept Type Parameters 848

20.8 Wrap-Up 852

21 Custom Generic Data Structures 856

21.1 Introduction 857

21.2 Self-Referential Classes 858

21.3 Dynamic Memory Allocation 858

21.4 Linked Lists 859

21.4.1 Singly Linked Lists 859

21.4.2 Implementing a Generic List Class 860

21.4.3 Generic Classes ListNode and List 863

21.4.4 Class ListTest 863

21.4.5 List Method insertAtFront 865

21.4.6 List Method insertAtBack 866

21.4.7 List Method removeFromFront 866

21.4.8 List Method removeFromBack 867

21.4.9 List Method print 868

21.4.10 Creating Your Own Packages 868

21.5 Stacks 873

21.6 Queues 876

Table of Contents

21.7 Trees 878

21.8 Wrap-Up 885

22 JavaFX Graphics and Multimedia 910

22.1 Introduction 911

22.2 Controlling Fonts with Cascading Style Sheets (CSS) 912

22.2.1 CSS That Styles the GUI 912

22.2.2 FXML That Defines the GUI Introduction to XML Markup 915

22.2.3 Referencing the CSS File from FXML 918

22.2.4 Specifying the VBoxs Style Class 918

22.2.5 Programmatically Loading CSS 918

22.3 Displaying Two-Dimensional Shapes 919

22.3.1 Defining Two-Dimensional Shapes with FXML 919

22.3.2 CSS That Styles the Two-Dimensional Shapes 922

22.4 Polylines, Polygons and Paths 924

22.4.1 GUI and CSS 925

22.4.2 PolyShapesController Class 926

22.5 Transforms 929

22.6 Playing Video with Media, MediaPlayer and MediaPlayer 931

22.6.1 VideoPlayer GUI 932

22.6.2 VideoPlayerController Class 934

22.7 Transition Animations 938

22.7.1 TransitionAnimations.fxml 938

22.7.2 TransitionAnimationsController Class 940

22.8 Timeline Animations 944

22.9 Frame-by-Frame Animation with AnimationTimer 947

22.10 Drawing on a Canvas 949

22.11 Three-Dimensional Shapes 954

22.12 Wrap-Up 958

Table of Contents

23 Concurrency 973

23.1 Introduction 974

23.2 Thread States and Life Cycle 976

23.2.1 New and Runnable States 977

23.2.2 Waiting State 977

23.2.3 Timed Waiting State 977

23.2.4 Blocked State 977

23.2.5 Terminated State 977

23.2.6 Operating-System View of the Runnable State 978

23.2.7 Thread Priorities and Thread Scheduling 978

23.2.8 Indefinite Postponement and Deadlock 979

23.3 Creating and Executing Threads with the Executor Framework 979

23.4 Thread Synchronization 983

23.4.1 Immutable Data 984

23.4.2 Monitors 984

23.4.3 Unsynchronized Mutable Data Sharing 985

23.4.4 Synchronized Mutable Data Sharing Making Operations Atomic 989

23.5 Producer/Consumer Relationship without Synchronization 992

23.6 Producer/Consumer Relationship: ArrayBlockingQueue 1000

23.7 (Advanced) Producer/Consumer Relationship with synchronized, wait, notify and notifyAll 1003

23.8 (Advanced) Producer/Consumer Relationship: Bounded Buffers 1009

23.9 (Advanced) Producer/Consumer Relationship: The Lock and Condition Interfaces 1017

23.10 Concurrent Collections 1024

23.11 Multithreading in JavaFX 1026

Table of Contents

23.11.1 Performing Computations in a Worker Thread: Fibonacci Numbers 1027

23.11.2 Processing Intermediate Results: Sieve of Eratosthenes 1032

23.12 sort/parallelSort Timings with the Java SE 8 Date/Time API 1038

23.13 Java SE 8: Sequential vs. Parallel Streams 1041

23.14 (Advanced) Interfaces Callable and Future 1043

23.15 (Advanced) Fork/Join Framework 1048

23.16 Wrap-Up 1048

24 Accessing Databases with JDBC 1060

24.1 Introduction 1061

24.2 Relational Databases 1062

24.3 A books Database 1063

24.4 SQL 1067

24.4.1 Basic SELECT Query 1068

24.4.2 WHERE Clause 1068

24.4.3 ORDER BY Clause 1070

24.4.4 Merging Data from Multiple Tables: INNER JOIN 1072

24.4.5 INSERT Statement 1073

24.4.6 UPDATE Statement 1074

24.4.7 DELETE Statement 1075

24.5 Setting Up a Java DB Database 1076

24.5.1 Creating the Chapters Databases on Windows 1077

24.5.2 Creating the Chapters Databases on macOS 1078

24.5.3 Creating the Chapters Databases on Linux 1078

24.6 Connecting to and Querying a Database 1078

24.6.1 Automatic Driver Discovery 1080

24.6.2 Connecting to the Database 1080

24.6.3 Creating a Statement for Executing Queries 1081

Table of Contents

24.6.4	Executing a Query	1081
24.6.5	Processing a Query's ResultSet	1082
24.7	Querying the books Database	1083
24.7.1	ResultSetTableModel Class	1083
24.7.2	DisplayQueryResults App's GUI	1090
24.7.3	DisplayQueryResultsController Class	1090
24.8	RowSet Interface	1095
24.9	PreparedStatement	1098
24.9.1	AddressBook App That Uses PreparedStatement	1099
24.9.2	Class Person	1099
24.9.3	Class PersonQueries	1101
24.9.4	AddressBook GUI	1104
24.9.5	Class AddressBookController	1105
24.10	Stored Procedures	1110
24.11	Transaction Processing	1110
24.12	Wrap-Up	1111
25	Introduction to JShell: Java 9's REPL for Interactive Java	1119
25.1	Introduction	1120
25.2	Installing JDK 9	1122
25.3	Introduction to JShell	1122
25.3.1	Starting a JShell Session	1123
25.3.2	Executing Statements	1123
25.3.3	Declaring Variables Explicitly	1124
25.3.4	Listing and Executing Prior Snippets	1126
25.3.5	Evaluating Expressions and Declaring Variables Implicitly	1128
25.3.6	Using Implicitly Declared Variables	1128
25.3.7	Viewing a Variable's Value	1129

Table of Contents

25.3.8	Resetting a JShell Session	1129
25.3.9	Writing Multiline Statements	1129
25.3.10	Editing Code Snippets	1130
25.3.11	Exiting JShell	1133
25.4	Command-Line Input in JShell	1133
25.5	Declaring and Using Classes	1134
25.5.1	Creating a Class in JShell	1135
25.5.2	Explicitly Declaring Reference-Type Variables	1135
25.5.3	Creating Objects	1136
25.5.4	Manipulating Objects	1136
25.5.5	Creating a Meaningful Variable Name for an Expression	1137
25.5.6	Saving and Opening Code-Snippet Files	1138
25.6	Discovery with JShell Auto-Completion	1138
25.6.1	Auto-Completing Identifiers	1139
25.6.2	Auto-Completing JShell Commands	1140
25.7	Exploring a Classs Members and Viewing Documentation	1140
25.7.1	Listing Class Maths static Members	1141
25.7.2	Viewing a Methods Parameters	1141
25.7.3	Viewing a Methods Documentation	1142
25.7.4	Viewing a public Fields Documentation	1142
25.7.5	Viewing a Classs Documentation	1143
25.7.6	Viewing Method Overloads	1143
25.7.7	Exploring Members of a Specific Object	1144
25.8	Declaring Methods	1146
25.8.1	Forward Referencing an Undeclared MethodDeclaring Method displayCubes	1146
25.8.2	Declaring a Previously Undeclared Method	1146
25.8.3	Testing cube and Replacing Its Declaration	1147
25.8.4	Testing Updated Method cube and Method displayCubes	1147

Table of Contents

25.9 Exceptions	1148
25.10 Importing Classes and Adding Packages to the CLASSPATH	1149
25.11 Using an External Editor	1151
25.12 Summary of JShell Commands	1153
25.12.1 Getting Help in JShell	1154
25.12.2 /edit Command: Additional Features	1155
25.12.3 /reload Command	1155
25.12.4 /drop Command	1156
25.12.5 Feedback Modes	1156
25.12.6 Other JShell Features Configurable with /set	1158
25.13 Keyboard Shortcuts for Snippet Editing	1159
25.14 How JShell Reinterprets Java for Interactive Use	1159
25.15 IDE JShell Support	1160
25.16 Wrap-Up	1160
Chapters on the Web	1176
A Operator Precedence Chart	1177
B ASCII Character Set	1179
C Keywords and Reserved Words	1180
D Primitive Types	1181
E Using the Debugger	1182
E.1 Introduction	1183
E.2 Breakpoints and the run, stop, cont and print Commands	1183
E.3 The print and set Commands	1187
E.4 Controlling Execution Using the step, step up and next Commands	1189
E.5 The watch Command	1191

Table of Contents

E.6 The clear Command 1193

Appendices on the Web 1197

Index 1199

Symbols

Numerics

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

Table of Contents

V

W

X

Y

Z